



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/847,535	05/01/2001	Barry Bond	MSI-665US	4017

22801 7590 06/05/2006
LEE & HAYES PLLC
421 W RIVERSIDE AVENUE SUITE 500
SPOKANE, WA 99201

EXAMINER

STEVENS, THOMAS H

ART UNIT	PAPER NUMBER
----------	--------------

2123

DATE MAILED: 06/05/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/847,535

Applicant(s)

BOND ET AL.

Examiner

Thomas H. Stevens

Art Unit

2123

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 10 March 2006.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-42, 45 and 46 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-42, 45 and 46 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☐ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date 9/12/05; 12/29/05

- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: _____


WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER

DETAILED ACTION

1. Claims 1-42 and 45-46 were examined.

Section I: Prosecution Reopened

2. In view of the appeal brief filed on 03/10/2006 and it was determined that the limitations regarding native and non-native were broad terms subject to interpretation. Therefore the examiner is setting forth a new ground of rejection. PROSECUTION IS HEREBY REOPENED. A new ground of rejection is set forth below.

To avoid abandonment of the application, appellant must exercise one of the following two options:

(1) file a reply under 37 CFR 1.111 (if this Office action is non-final) or a reply under 37 CFR 1.113 (if this Office action is final); or,

(2) initiate a new appeal by filing a notice of appeal under 37 CFR 41.31 followed by an appeal brief under 37 CFR 41.37. The previously paid notice of appeal fee and appeal brief fee can be applied to the new appeal. If, however, the appeal fees set forth in 37 CFR 41.20 have been increased since they were previously paid, then appellant must pay the difference between the increased fees and the amount previously paid.

A Supervisory Patent Examiner (SPE) has approved of reopening prosecution by signing below:

Section II: Non-Final Rejection

Claim Interpretation

3. Office personnel are to give claims their "**broadest reasonable interpretation**" in light of the supporting disclosure. In re Morris, 127 F.3d 1048, 1054-55, 44 USPQ2d

1023, 1027-28 (Fed. Cir. 1997). Limitations appearing in the specification but not recited in the claim are not read into the claim. In re Prater, 415 F.2d 1393, 1404-05, 162 USPQ 541, 550-551 (CCPA 1969). See *also In re Zletz, 893 F.2d 319, 321-22, 13 USPQ2d 1320, 1322 (Fed. Cir. 1989) ("During patent examination the pending claims must be interpreted as broadly as their terms reasonably allow") The reason is simply that during patent prosecution when claims can be amended, ambiguities should be recognized, scope and breadth of language explored, and clarification imposed An essential purpose of patent examination is to fashion claims that are precise, clear, correct, and unambiguous. Only in this way can uncertainties of claim scope be removed, as much as possible, during the administrative process. The Office interprets the kernel emulator process as both software/hardware since the disclosure is unclear of what an emulator truly is. Furthermore, according to Microsoft Press Computer User's Dictionary, 1998 edition, a kernel is defined (pg. 201) as the core of an operating system—the portion of the system that manages memory, files and peripheral devices; maintains the time and date; launches application; and allocates system resources.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claims 1,3-6,9-13,15-17,19-28, 34,37-40,42,45-46 are rejected under 35 U.S.C. 102(b) as being anticipated by Scalzi et al (U.S. Patent Number 5,560,013), herein referred to as Scalzi.

As to Claims 1,3,4,13,15,16,34, 40 and 45, Scalzi teaches: a kernel emulator for non-native program modules, the emulator comprising: an interceptor configured to intercept kernel calls from non-native program modules (Figure 1, element 102 and description); a call-converter configured to convert non-native kernel calls intercepted by the interceptor into native kernel calls (Figure 1, element 103 and description); wherein the call-converter comprises: an instruction-translator configured to translate non-native CPU instructions into native CPU instructions (Figure 1, element 103 and description); an address-translator configured to translate addresses from non-native length into native length (Figure 3 and description, column 21, lines 42-48).

As to Claim 5, Scalzi teaches: an emulator as recited in claim 1, wherein the call-converter comprises an argument-converter configured to convert non-native argument format into native argument format (Figure 3 and description, column 21, lines 42-48).

As to Claim 6,17 and 37, Scalzi teaches: an emulator as recited in claim 1, wherein the call-converter comprises a translator configured to translate words from non-native word size into native word size (Figure 3 and description, column 21, lines 42-48).

As to Claims 9 and 19, Scalzi teaches: an emulator as recited in claim 1 further comprising a shared-memory manager configured to synchronize a native shared data structure with a non-native shared data structure (column 18, lines 46-65).

As to Claims 10, 20-25, Scalzi teaches: an emulator as recited in claim 1 further comprising a shared-memory manager configured to manage memory space that is accessible to both native and non-native program modules, wherein the shared-memory manager maps versions of process shared data structures (SDSs) and versions of thread shared data structures (SDSs) between native and non-native program modules (column 1, lines 23-25, column 2, lines 54-62, column 3, lines 9-22, column 5, lines 4-16) wherein the processes shared data structures include environmental conditions of the machine and the thread shared data structure is encompassed by the control programs.

As to Claims 11, 42 and 46, Scalzi teaches: an operating system on a computer-readable medium, comprising: a native kernel configured to receive calls from native program modules (column 17, line 56-57); a kernel emulator as recited in claim 1 configured to receive calls from non-native program modules (Figure 1, element 102 and description).

As to Claim 12, Scalzi teaches: an operating system on a computer-readable medium, comprising: a native kernel configured to receive calls from native APIs (column 17, line 56-57); a kernel emulator as recited in claim 1 configured to receive calls from non-native APIs (Figure 1, element 102 and description).

As to Claims 26,27,28,38,39 Scalzi teaches: a computer comprising one or more computer-readable media having computer-executable instructions that, when executed by the computer, perform the method as recited in claim 13 (Figure 1, element 102 and

description) and an operating system embodied on a computer readable medium (column 3, lines 32-39).

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

7. The factual inquiries set forth in *Graham v. John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of

the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

8. Claims 1-42, 45 and 46 are rejected under 35 U.S.C. 103 (a) as being unpatentable over Scalzi et al. (US Patent 5,560,013), (hereafter Scalzi). Scalzi teaches an emulator control program that manages target processors.

Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to utilize the management of dynamic changes within the source machine storage in multitasking source operation of Scalzi because Scalzi teaches a method to improve system performance of linkage to, and data access by, that code (Scalzi: column 4, lines 5-6).

As to Claims 1,3,4,13,15,16,34, 40 and 45, Scalzi teaches: a kernel emulator for non-native program modules, the emulator comprising: an interceptor configured to intercept kernel calls from non-native program modules (Figure 1, element 102 and description); a call-converter configured to convert non-native kernel calls intercepted by the interceptor into native kernel calls (Figure 1, element 103 and description); wherein the call-converter comprises: an instruction-translator configured to translate non-native CPU instructions into native CPU instructions (Figure 1, element 103 and description);

an address-translator configured to translate addresses from non-native length into native length (Figure 3 and description, column 21, lines 42-48).

As to Claim 5, Scalzi teaches: an emulator as recited in claim 1, wherein the call-converter comprises an argument-converter configured to convert non-native argument format into native argument format (Figure 3 and description, column 21, lines 42-48).

As to Claim 6, 17 and 37, Scalzi teaches: an emulator as recited in claim 1, wherein the call-converter comprises a translator configured to translate words from non-native word size into native word size (Figure 3 and description, column 21, lines 42-48).

As to Claims 9 and 19, Scalzi teaches: an emulator as recited in claim 1 further comprising a shared-memory manager configured to synchronize a native shared data structure with a non-native shared data structure (column 18, lines 46-65).

As to Claims 10, 20-25, Scalzi teaches: an emulator as recited in claim 1 further comprising a shared-memory manager configured to manage memory space that is accessible to both native and non-native program modules, wherein the shared-memory manager maps versions of process shared data structures (SDSs) and versions of thread shared data structures (SDSs) between native and non-native program modules (column 1, lines 23-25, column 2, lines 54-62, column 3, lines 9-22, column 5, lines 4-16) wherein the processes shared data structures include environmental conditions of the machine and the thread shared data structure is encompassed by the control programs.

As to Claims 11, 42 and 46, Scalzi teaches: an operating system on a computer-readable medium, comprising: a native kernel configured to receive calls from native program modules (column 17, line 56-57); a kernel emulator as recited in claim 1 configured to receive calls from non-native program modules (Figure 1, element 102 and description).

As to Claim 12, Scalzi teaches: an operating system on a computer-readable medium, comprising: a native kernel configured to receive calls from native APIs (column 17, line 56-57); a kernel emulator as recited in claim 1 configured to receive calls from non-native APIs (Figure 1, element 102 and description).

As to Claims 26,27,28,38,39 Scalzi teaches: a computer comprising one or more computer-readable media having computer-executable instructions that, when executed by the computer, perform the method as recited in claim 13 (Figure 1, element 102 and description) and an operating system embodied on a computer readable medium (column 3, lines 32-39).

9. Claims 2 and 14 is rejected under 35 U.S.C. 103(a) as being unpatentable over Scalzi as applied to Claim 1 above, and further in view of Franz (Michael Franz, "Emulating an Operating System on Top of Another" Software - Practice and Experience. Vol. 23, No. 6, June 1993, pp. 677-692), herein referred to as Franz. Scalzi and Franz are analogous art since they both teach computer-based operating systems.

It would have been obvious at the time the invention was made to utilize the translator configured to translate a non-native paradigm for passing parameters into a

native paradigm for passing parameters of Franz in the non-native CPU instructions into native CPU instructions of Scalzi because Franz teaches that incompatible paradigms are one of the major hurdles that needs to be overcome in order to emulate one operating system on top of the other (Franz: page 678, "Barriers to Portability").

As to Claims 2 and 14, Scalzi teaches: a call converter comprising an instruction-translator configured to translate non-native CPU instructions into native CPU instructions (Figure 1, element 103 and description).

Franz teaches: an emulator as recited in claim 1, wherein the call-converter comprises a translator configured to translate a non-native paradigm for passing parameters into a native paradigm for passing parameters (page 681, paragraph 4, "boot loader") as a method to enable the emulation of one operating system on top of the other. Franz teaches that incompatible paradigms are one of the major hurdles that needs to be overcome in order to emulate one system on top of the other (page 678, "Barriers to Portability").

10. Claims 7, 8, 18, 35 and 41 are rejected under 35 U.S.C. 103(a) as being unpatentable over Scalzi as applied to Claim 1 above, and further in view of Duvall et al (U.S. Patent Number 4,742,447), herein referred to as Duvall. Both Scalzi and Duvall are analogous art since both teach virtual machines.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to utilize the virtual machines running different operating systems in a UNIX environment of Duval in the translating non-native CPU instructions

Art Unit: 2123

into native CPU instructions of Scalzi because Duval teaches a method which allows the personal computer operating in an emulation mode to directly utilize an application program (Duval: column 3, lines 31-34).

As to Claims 7, 8, 18, 35 and 41, Scalzi teaches a block of memory configured as a range addressable by non-native program modules (Figure 1, element 109) and memory space that is accessible to both native and non-native modules (Figure 1, "Target Real" wherein the target storage is part of the physical storage of the native platform).

As to Claims 35 and 41, Scalzi teaches: translating non-native CPU instructions into native CPU instructions (Figure 1, element 103 and description); translating addresses from non-native length into native length (Figure 3 and description, column 21, lines 42-48).

Duvall teaches a memory constrainer configured to limit addressable memory to a range addressable by non-native program modules (column 4, lines 43-46, column 6, lines 25-29, column 9, lines 20-25) or a memory manger configured to manage memory space that is accessible to both native and non-native program modules (Figure 1, element 13, column 9, lines 21-25). Duvall teaches this method for virtual machines running different operating systems in a UNIX environment (column 8, line 65-column 9, line 3), virtual memory (Figure 3) and binary address translation (column 9, lines 26-38).

Art Unit: 2123

11. Claim 36 is rejected under 35 U.S.C. 103(a) as being unpatentable over Scalzi and Duvall as applied to claim 35 above, and further in view of Franz. Scalzi, Duval and Franz are analogous art since they all teach internal computer software.

It would have been obvious at the time the invention was made to utilize the software interface of Franz in the call converter in Scalzi and Duvall because Franz teaches that incompatible paradigms are one of the major hurdles that needs to be overcome in order to emulate one operating system on top of the other (Franz: page 678, "Barriers to Portability").

As to Claim 36, Scalzi and Duval teaches: a call converter comprising an instruction-translator configured to translate non-native CPU instructions into native CPU instructions (Scalzi: Figure 1, element 103 and description).

Franz teaches: an emulator as recited in claim 1, wherein the call-converter comprises a translator configured to translate a non-native paradigm for passing parameters into a native paradigm for passing parameters (page 681, paragraph 4, "boot loader") as a method to enable the emulation of one operating system on top of the other. Franz teaches that incompatible paradigms are one of the major hurdles that needs to be overcome in order to emulate one system on top of the other (page 678, "Barriers to Portability").

Claims 29-33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Duvall (US Patent 47,42,447) in view of McCoy et al (U.S. Patent Number 5,036,484),

Art Unit: 2123

herein referred to as McCoy. Duvall and McCoy are analogous since they both teach emulation events on a computer.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to utilize emulation program of McCoy in the virtual machine of Duval because McCoy teaches a system that allows the personal computer operating in an emulation mode to directly utilize an application program (Duval: column 3, lines 32-35).

As to Claim 29, Duvall teaches: a method comprising: if the initiating program is non-native: limiting available memory to a range that is addressable by the non-native program module (column 4, lines 43-46, column 6, lines 25-29, column 9, lines 20-25); establishing non-native a version of a shared memory data structure that may be synchronized with a native version of the same shared memory data structure (column 5, lines 45-51, column 6, lines 25-29).

Duvall further teaches the data in a segment of virtual memory is created as a result of an application program being run (column 5, lines 52-55). While this implies that must be some determination as to whether a program is native or non-native allowing for the segment in virtual memory to be created, Duvall does not expressly teach determining whether an initiating program module is a native or non-native.

McCoy teaches determining whether an initiating program module is a native or non-native (Figure 3a, element 36a, column 5, lines 40-48) in a system that emulates a host program in a PC environment and translates host data to PC format by the emulation program (column 5, lines 28-31), allowing the system to know whether to

Art Unit: 2123

perform a function of the native system or perform a function of the non-native system which includes the translation of code (Figure 3a, element 31a and column 5, lines 40-48) since in the emulation systems of the prior art, when operating in emulation mode, the native system is incapable of performing functions other than those of the terminal which is being emulated. Therefore, the functions of the personal computer are not available in the emulation mode (column 1, lines 32-39).

As to Claim 30, Duvall and McCoy teach: a method as recited in claim 29 further comprising: intercepting kernel calls from the non-native program module; converting the intercepted non-native kernel calls into native kernel calls (Duvall: column 9, lines 26-41, McCoy, Figure 3a, elements 31a and 36a).

As to Claim 31, Duvall and McCoy teach: a method as recited in claim 29 further comprising emulating a non-native kernel for which kernel calls from the non-native program module are intended (Duvall: column 8, line 62-column 9, line 6, McCoy: Figure 3, element 35).

As to Claim 32, Duvall and McCoy teach: a computer comprising one or more computer-readable media having computer-executable instructions that, when executed by the computer, perform the method as recited in claim 29 (Duvall: column 8, lines 50-54, line 62-column 9, line 3, McCoy, column 4, lines 30-33).

As to Claim 33, Duvall and McCoy teach: a computer-readable medium having computer-executable instructions that, when executed by a computer, performs the method as recited in claim 29 (Duvall: column 8, lines 50-54, line 62-column 9, line 3, McCoy, column 4, lines 30-33).

Section III: Response to Applicants' Arguments (from the Appeal Brief)

Claims 1-28, 40-46

Applicants state the Scalzi (U.S. Patent 5,560,013 (1996)) reference doesn't disclose kernel of an operating system. According to Microsoft Press Computer User's Dictionary, 1998 edition, a kernel is defined (pg. 201) as *the core of an operating system—the portion of the system that manages memory, files and peripheral devices; maintains the time and date; launches application; and allocates system resources.* Scalzi defines the core operating systems as the "target system" that is defined (abstract) as *efficiently stores the translated executable source programs by actually storing only one copy of any source program, regardless of the number of source address spaces in which the source program exists. The target system efficiently manages dynamic changes in the source machine storage, accommodating the nature of a preemptive, multitasking source operating system. The target system preserves the security and data integrity for the source programs on a par with their security and data integrity obtainable when executing in source processors (i.e., having the source architecture as their native architecture). The target computer execution maintains source-architected logical separations between programs and data executing in different source address spaces—without a need for the target system to be aware of the source virtual address spaces.* Summarily, both terms are defined as the managing source of the computer (highlight emphasized), thus equating the same purpose.

Applicants state Scalzi doesn't disclose interception of a kernel call or kernel emulation. Based on the latter analysis of equivalence between kernel and target system examiner argues the kernel call is inherent for the simple reason that if a translation function is an event between nativity and non-native functions then a call is understood as an integral part to the translation function (Scalzi: column 5, lines 9-11, "instruction translation"; column 5, lines 38-40). Furthermore, an interception must take place if the such an event between the native and non-native functions is to exist.

Claims 34-39

Scalzi states a method of emulating a method for executing individual source instructions in a target processor to execute source programs (column 24, lines 59-61). The translating the Applicants are referring to what is believed to be inherent to the nature of the events between the native and non-native functions. Coupled with the rebuttal in claim 1, Scalzi's abstract states the "*translated executable source programs*" thus teaching a translation of source code between the native and non-native functions.

Claims 29,30-33

The Duvall reference's discloses the range or limit of the virtual memory, in this case 1 terrabyte, which anyone with ordinary skill in the art would know is addressable. Applicants are correct to Duval's lack of teaching of an emulator; however, the rejection is coupled to the McCoy reference that discloses an emulator. One cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA

1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). Furthermore, examiner agrees McCoy does not disclose such an emulation program since "emulation program" was not disclosed in claims 29-23.

No Motivation to Combine References

In response to applicants' argument that the Office has not provided any objective evidence showing why OOSA would be motivated to combine the teachings of the two references, the fact that applicant has recognized another advantage which would flow naturally from following the suggestion of the prior art cannot be the basis for patentability when the differences would otherwise be obvious. See *Ex parte Obiaya*, 227 USPQ 58, 60 (Bd. Pat. App. & Inter. 1985).

Correspondence Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Mr. Tom Stevens whose telephone number is 571-272-3715, Monday-Friday (8:00 am- 4:30 pm EST).

If attempts to reach the examiner by telephone are unsuccessful, please contact examiner's supervisor Mr. Paul Rodriguez 571-272-3753. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.


Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status

Art Unit: 2123

information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Answers to questions regarding access to the Private PAIR system, contact the Electronic Business Center (EBC) (toll-free (866-217-9197)).

May 23, 2006

TS


WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER